

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN
THÔNG**



BÁO CÁO THỰC TẬP CƠ SỞ

ĐỀ TÀI

XÂY DỰNG WEBSITE THƯƠNG MẠI ĐIỆN TỬ

Giảng viên hướng dẫn : Kim Ngọc Bách

Họ và tên : Nguyễn Trọng Nam

Mã sinh viên : B22DCAT204

Lớp : E22CQCN02-B

I. Code backend xác thực người dùng và giải thích về cách hoạt động của chúng

(Các module được sử dụng:

```
const db = require('./config/db');  
// Kết nối database MySQL  
const bcrypt = require('bcrypt');  
// Mã hóa mật khẩu  
const jwt = require('jsonwebtoken');  
// Tạo và xác thực JWT token  
const _const = require('./config/constant');  
// Các hằng số như JWT_ACCESS_KEY  
const nodemailer = require('nodemailer');  
// Gửi email  
const crypto = require('crypto');  
// Tạo token ngẫu nhiên )
```

1. Chức năng đăng ký (registerUser)

Khi người dùng đăng ký tài khoản mới:

- Mật khẩu được mã hóa bằng bcrypt để đảm bảo an toàn
- Hệ thống kiểm tra xem email đã tồn tại chưa
- Nếu email chưa tồn tại, thông tin người dùng được lưu vào bảng users
- Trả về thông tin người dùng (không bao gồm mật khẩu) nếu thành công

2. Chức năng đăng nhập (login)

Quá trình đăng nhập:

- Kiểm tra email trong cơ sở dữ liệu
- So sánh mật khẩu đã hash với mật khẩu người dùng nhập
- Nếu xác thực thành công, tạo JWT token với thời hạn 30 ngày
- Token được gửi về client qua header 'Authorization' và trong response body
- Client lưu token này để sử dụng cho các request yêu cầu xác thực

3. Chức năng quên mật khẩu (forgotPassword)

Khi người dùng quên mật khẩu:

- Tạo token ngẫu nhiên bằng crypto
- Lưu token vào bảng password_reset_tokens với thời hạn 1 giờ
- Cấu hình SMTP server (sử dụng Brevo/SendinBlue) để gửi email
- Gửi email chứa link reset mật khẩu đến người dùng
- Link có dạng: `http://localhost:3000/reset-password/{token}`

4. Chức năng đặt lại mật khẩu (resetPassword)

Khi người dùng nhấp vào link reset:

- Xác thực token từ request
- Kiểm tra token có tồn tại và còn hạn không
- Nếu hợp lệ, mã hóa mật khẩu mới và cập nhật vào database
- Xóa token đã sử dụng để đảm bảo an toàn

- Thông báo kết quả về client

5. Xử lý lỗi

Mỗi chức năng đều có khối try-catch để xử lý lỗi:

- Lỗi được ghi log qua `console.error()`
- Trả về status code 500 và thông báo lỗi cho client
- Giúp debug và xử lý sự cố dễ dàng hơn

6. Kết nối với Client

File này xử lý các request API từ phía client như:

- Đăng ký tài khoản mới
- Đăng nhập và nhận JWT token
- Yêu cầu reset mật khẩu
- Xác nhận đổi mật khẩu mới

Đây là một phần thiết yếu của hệ thống backend, đảm bảo tính bảo mật và xác thực cho toàn bộ ứng dụng TrendyWear.